YAWL
Installation Manual

Beta – 8 Release

Document Control

Date	Author	Version	Change
11 Aug 2004	Sean Kneipp	0.1	Initial Draft
18 Aug 2004	Sean Kneipp	0.2	Additional Troubleshooting
23 Aug 2004	Sean Kneipp	1.0	Final
4 Feb 2005	Guy Redding	1.1	Added YAWLXForms.war detail
4 Feb 2005	Lachlan Aldred	1.2	Updated issues and 3rd party software acknowledgements.
12 Apr 2005	Lachlan Aldred	1.3	Added comments on JVM & Tomcat compatibility.
13 May 2005	Petia Wohed	1.4	Corrections
16 May 2005	Lachlan Aldred	1.5	Added note about SMSInvoker.
28/10/2005	Lachlan Aldred	1.6	Postgres installation. Admintool.
11/12/2005	Michael Adams	1.7	Added Worklet Service
13 Feb 2006	Lachlan Aldred	1.8	Refinements to Postgres installation.
24 Feb 2006	Lachlan Aldred	1.9	General refinements
11 Oct 2006	Michael Adams	1.99	Revised for Beta 8

Preface

This manual contains instructions for installing YAWL version Beta 8.

How to Use This Manual

The Requirements & Installation chapter provides installation instructions for the two software requirements, Java 2 SDK and Tomcat, and then continues on to provide detailed installation instructions for the YAWL Engine.

Contents

Welcome to YAWL

What is YAWL?

Based on a rigorous analysis of existing workflow management systems and workflow languages, we have developed a new workflow language: YAWL (Yet Another Workflow Language). To identify the differences between the various languages, we have collected a fairly complete set of [workflow patterns](). Based on these patterns we have evaluated several workflow products and detected considerable differences in expressive power. Languages based on Petri nets perform better when it comes to state-based workflow patterns. However, some patterns (e.g. involving multiple instances, complex synchronizations or non-local withdrawals) are not easy to map onto (high-level) Petri nets. This inspired us to develop a new language by taking Petri nets as a starting point and adding mechanisms to allow for a more direct and intuitive support of the workflow patterns identified.

YAWL can be considered a very powerful workflow language, built upon experiences with languages supported by contemporary workflow management systems. While not a commercial language itself it encompasses these languages, and, in addition, has a formal semantics. Such an approach is in contrast with e.g. WfMC's XPDL which takes commonalities between various languages as a starting point and does not have formal semantics. Its design hopefully allows YAWL to be used for the purposes of the study of expressiveness and interoperability issues.

At this stage YAWL supports the control-flow perspective and the data perspective. However, it is important that relations with the resource perspective are investigated and formalized. For YAWL to be more applicable in the area of web services and Enterprise Application Integration, it is also desirable that support for communication patterns is built-in.

Obtaining the Latest Version of the YAWL Engine

As new versions of the YAWL Engine are released to the public, they will be available for download at the YAWL website:

http://www.yawl-system.com

Developers interested in obtaining the source code for the YAWL Engine, can download the files from here:

http://sourceforge.net/projects/yawl

The YAWL Project

For more information and progress on the YAWL project, visit the YAWL Homepage, hosted by Queensland University of Technology's, Faculty of Information Technology.

Requirements

- Java 2, Standard Edition (J2SE), 1.4.2_10

- Apache Tomcat 5.0.28

- PostgreSQL – 8: If you wish to use database persistence.

- YAWL Beta – 8 has been tested under Windows NT/XP and Linux.

NOTE:

We have tested the platform with several combinations of Java and Tomcat, and found issues related to the submission of forms for versions: Java 5.0, and Tomcat 5.0.16, and Tomcat 5.5.x.

Java 2 SDK

The Java 2 Software Development Kit, Standard Edition, v1.4.2_10 (the last version of Java 2), can be downloaded from Sun Microsystems Java website: http://java.sun.com/

1. Execute the downloaded Installer.

2. Once the installer has completed, you need to set up the `JAVA_HOME` environment variable for your operating system, and point it to the base directory of your JDK e.g., `'C:\j2sdk1.4.2_10'`.

    Instructions for setting up `JAVA_HOME` under Windows XP

    i) Open the Control Panel.

    ii) Choose **System** from the Control Panel.

    iii) Go to the Advanced tab of the Control Panel and choose '**Environment Variables**' (see Figure 1: **Environment Variables Menu**).

Figure 1: Environment Variables Menu

iv) Under the System Variables section, locate the "JAVA_HOME" variable and choose **Edit**. If JAVA_HOME is not in the list of System Variables, choose '**New'** (see Figure 2: **Locate the JAVA_HOME System variable**)**.**
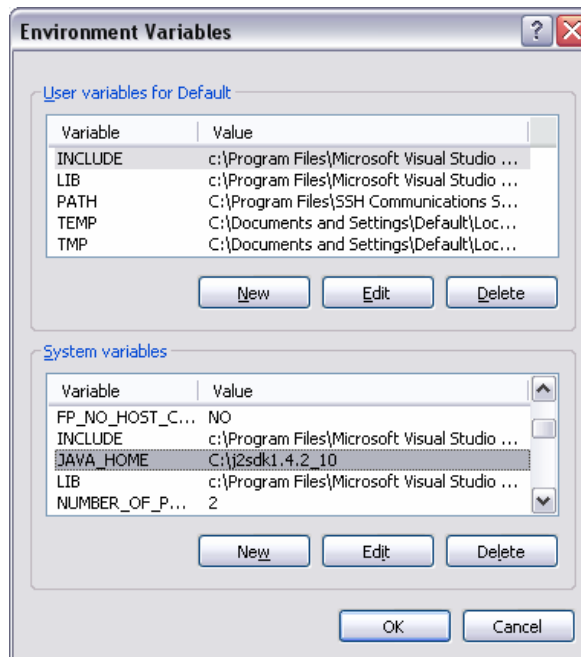

Figure 2: Locate the JAVA_HOME System variable

v) In the Variable name field type in JAVA_HOME. In the Variable value type in the location of the folder you installed the Java 2 Software Development Kit, eg. C:\j2sdk1.4.2_10 (see Figure 3: **Setting up the JAVA_HOME variable.**
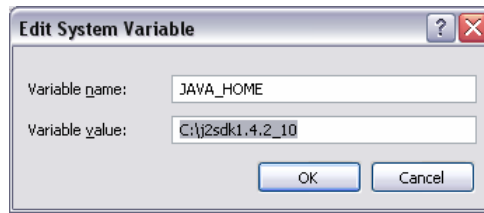
Figure 3: Setting up the JAVA_HOME variable.

vi) Click OK, OK, OK. The JAVA_HOME environment variable has now been set up.

Apache Tomcat

Download Tomcat, from the Tomcat website: http://jakarta.apache.org/tomcat/ ▪, or click on the download link below:

1. Execute the downloaded Installer.

2. Once the installer has completed, you need to set up the following environment variable into your operating system:

   CATALINA_HOME and point it to the root directory of Tomcat (e.g., 'C:\Program Files\Apache Software Foundation\Tomcat 5.0').

   Instructions for setting up CATALINA_HOME under Windows XP

   i) Go to the Environment Variables in the Control Panel. *See section i) – iii) of previous section, Instructions for JAVA_HOME.*

   ii) Under the System Variables section, locate the CATALINA_HOME variable and choose **Edit**. If CATALINA_HOME is not in the list of System Variables, choose **New** (see Figure 4: **Check for the presence of the CATALINA_HOME System variable**).
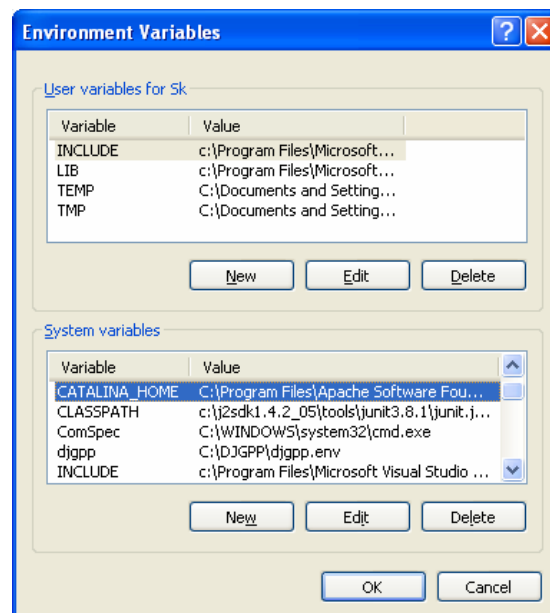
iii) In the Variable name field type in CATALINA_HOME. In the Variable value type in the location of the folder you installed the Tomcat software, eg. `'C:\Program Files\Apache Software Foundation\Tomcat 5.0 (see` Figure 5: **Setting up the CATALINA_HOME System variable.**`).`
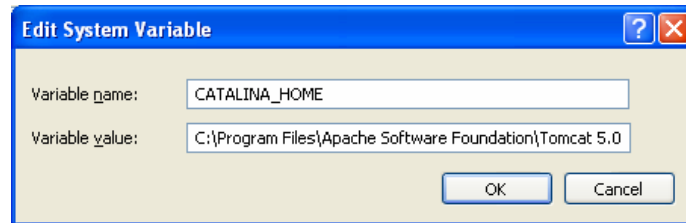

Figure 5: Setting up the CATALINA_HOME System variable.

iv) Click OK, OK, OK. Now the CATALINA_HOME environment variable has now been set up.

Starting Tomcat

Tomcat is a web-application hosting environment for hosting browser-based applications. YAWL being such requires a Servlet/JSP environment, such as Tomcat.

1. Launch Tomcat using the 'Monitor Tomcat' shortcut in the Start Menu.

   Start => Programs => Apache Tomcat 5.0 => Monitor Tomcat (Figure 6).

Figure 6 Launching Tomcat Monitor

2. Right-click on the Apache Tomcat icon in your System Tray and choose "Start Service" (Figure 7).
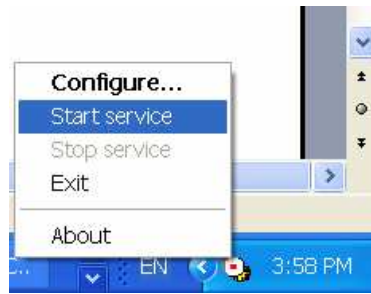

Figure 7: Starting Tomcat Service

If the Start Service option is not available, then the service has already been started. You will need to "Stop Service" then "Start Service".

Note: If you start the service, then right-click on the Apache Tomcat icon and choose "Exit", the service will remain Started even though you have exited the Tomcat Monitor.

3. Allow 15 to 20 seconds for the web service to initiate before continuing on with launching YAWL (see Launching the YAWL Engine).

About the Binary Distribution for YAWL

In the YAWL-Beta-8-Executable-WebServer.zip file are eight web application (.war) files:

yawl.war
'yawl.war' is the core workflow engine; it talks to the other components via XML messages sent over HTTP.
worklist.war
'worklist.war' is the worklist web application component. It serves HTML pages to multiple users, providing a view into the state of running workflow in the engine component.

yawlWSInvoker.war
'yawlWSInvoker.war' is the application integration component. It enables the engine to invoke online Web Services.

yawlSMSInvoker.war
'yawlSMSInvoker.war' is a custom YAWL service capable of sending and receiving SMS messages. To use it you will need to set up an account with a particular SMS company. More information can be found by clicking the link to the YAWL service from the administration page.

Note:
To use the SMS Invoker service you will first need to get some SMS credits from the company our tool uses to perform the SMS network connectivity (http://www.directsms.com.au). Then you will need to enter your account details

(username & password) into the YAWL tool. The entry form for this can be accessed by clicking on the link (http://localhost:8080/yawlSMSInvoker/ib) inside the administration page.

YAWLXForms.war
'YAWLXForms.war' is the forms processing component. It is used to generate and display forms for tasks as required in the YAWL engine.

admintool.war
This is the YAWL administration tool. You need this to add/edit the organisational resources and roles. If the process model is configured to allocate workitems to particular people then you will need to add them through this. URL: http://localhost:8080/admintool

workletService.war
The *Worklet Dynamic Process Selection and Exception Handling Service* is a Custom YAWL service that adds flexibility and exception handling capabilities for all standard YAWL processes. See the separate document *Worklet Custom Service Installation and User Manual – Beta 8* for more information about the worklet service.

timeService.war
The time-service is a YAWL compatible web service capable of providing timeout functionality to YAWL processes. This allows critical sections of YAWL processes to race against a defined deadline, which can result in compensation/escalation strategies if deadlines are not met.

The zip file also contains the following example YAWL Specifications, for example:
- § **BarnesAndNoble.xml:** written to invoke the Barnes and Noble web service and display the price of an arbitrary book using the ISBN.
- § **MakeRecordings.xml:** which is a nonsensical process based around recording songs for an album.

Also included in the binary distribution are the following two "jar" files:
`dom3-xercesImpl.jar` & `dom3-xml-apis.jar`: 3rd party libraries needed to support XForms display.

Installing YAWL

To install YAWL:

1. Extract the contents of '`YAWL-Beta-8-Executable-WebServer-Version.zip`' to a directory of your choice.

2. Copy each of the '.war' files directly into the 'webapps' directory inside your Tomcat installation (Figure 8)**.**
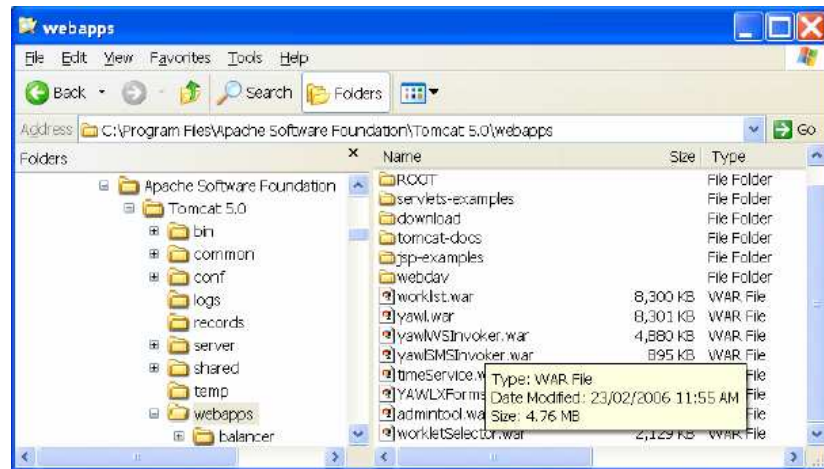
Figure 8: Installation of YAWL '.war' files

3. Copy **dom3-xercesImpl.jar** and **dom3-xml-apis.jar** from the YAWL-Beta-8-Executable-WebServer.zip file to %CATALINA_HOME%/common/endorsed.

4. Remove or backup any other **xercesImpl.jar** and **xml-apis.jar** possibly residing there. Alternatively you might install the extracted libraries as extensions in your JAVA_HOME.

Congratulations – YAWL installation is complete!

Enabling Persistence through PostgreSQL

You need to install PostgreSQL if you wish to use the full features of the YAWL product.   For instance the allocation of work items to different users (based on their role/s – known as the resource perspective) relies on a properly installed database. YAWL has persistence switched off by default, and therefore the installation of Postgres is in fact optional.

Installing PostgreSQL [Optional]

To install Postgres download the installer from the PostgreSQL website. Then double click the installer program for whatever platform and language requirements you have locally. For the Windows XP platform the Postgres installer will resemble the form **postgresql-x.x-int.msi.** Execute the MSI. In XP you are asked if you want to run it as a service:

1. Ensure that you are logged in to XP with Administrator privileges.
2. Choose to "install as a service" (see Figure 9: Choose "Install as Service" and an account password of your choosing. This will create a new user in the operating system, which, as stated, must not have admin privileges.).
3. Enter username and password for a user in XP that hosts the PostgreSQL service – this user should be underprivileged (see Figure 9).
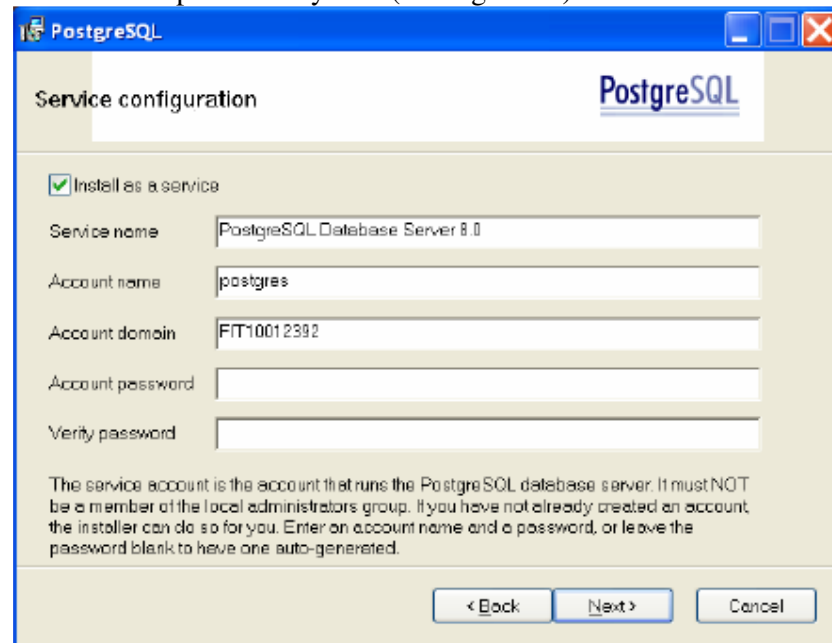4. Enter password "yawl" (see Figure 10).

Figure 9: Choose "Install as Service" and an account password of your choosing. This will create a new user in the operating system, which, as stated, must not have admin privileges.
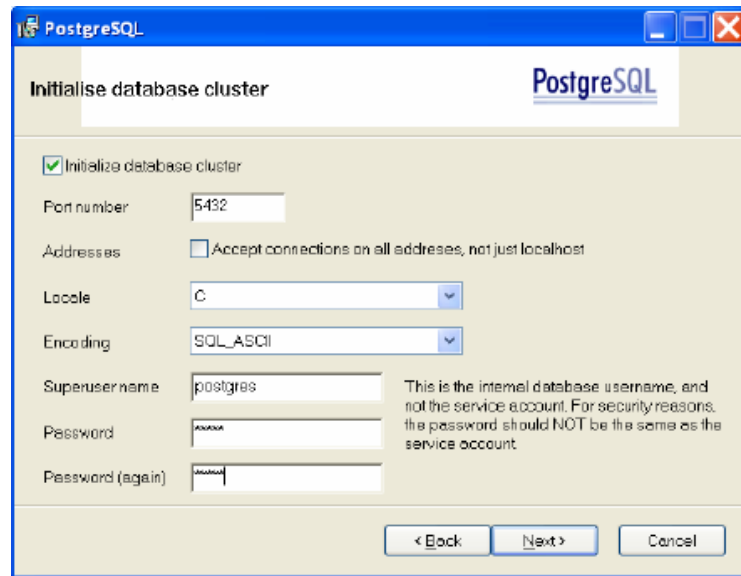
**Figure 10: YAWL is configured to use username "postgres",
password "yawl" to connect with the database[1].**

5. Click "Next" 4 times and Postgres is installed.
6. Now launch the pgAdmin tool (see Figure 11). Right click on the PostgreSQL Database Server and choose "connect". In the Connect to Server dialog box enter "yawl". Next right click on "Databases" and choose "New Database"
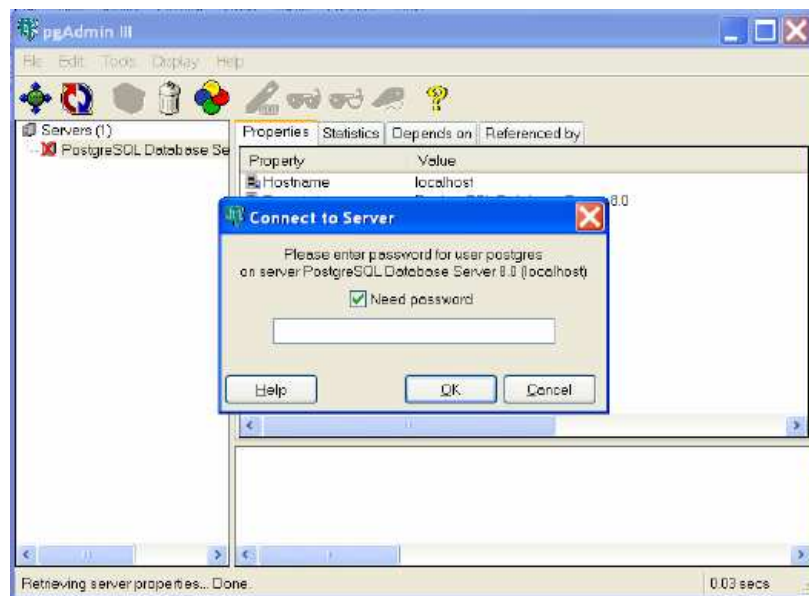

Figure 11: pgAdmin - the PostgreSQL Administration tool.

---

[1] If you wish to change the username/password combination that YAWL uses you will need to edit "`hibernate.connection.username`" and "`hibernate.connection.password`" of the "`hibernate.properties`" file. You'll also have to add the new user/password to Postgres.

7. In the next box choose "yawl" as the Database name, you can leave all the other fields blank. Click OK.

You do not need to create tables. The YAWL application can do this automatically when it is started.

Configuring YAWL Runtime to use PostgreSQL [Optional]

As mentioned previously, the YAWL system does not connect to a database by default, however configuring it to do so is not very difficult.

The easiest way to do this is to load the binary war files into Tomcat's "webapps" directory, and let Tomcat automatically load and extract their contents.

Stop Tomcat and open each of the following XML files:
```
%CATALINA_HOME%/webapps/yawl/WEB-INF/web.xml
%CATALINA_HOME%/webapps/worklist/WEB-INF/web.xml
%CATALINA_HOME%/webapps/admintool/WEB-INF/web.xml
%CATALINA_HOME%/webapps/workletService/WEB-INF/web.xml
```

In each of the files edit the context parameter "EnablePersistance" by changing the parameter's value from "false" to "true":
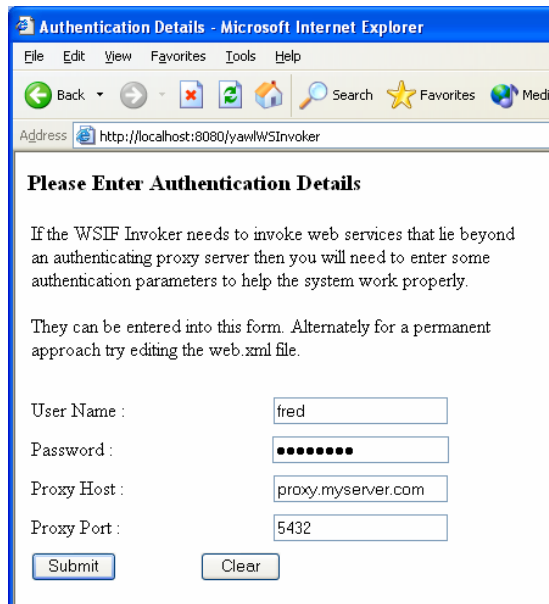
```
<context-param>
  <param-name>EnablePersistance</param-name>
  <param-value>true</param-value>
  <description>Set this value to 'true' to enable persistence and logging set to 'false' to
      disable</description>
</context-param>
```

Configuration for Proxy Servers [Optional]

If your computer resides inside of a network behind an authenticating proxy, in order to use the 'yawlWSInvoker' application to invoke services on the World Wide Web, some minor configuration is required.

1. Open up your web browser.

2. Type the following address into the address line and press enter:

   http://localhost:8080/yawlWSInvoker.

3. Fill in the following fields on the form, detailing your proxy server:

   - User Name – authenticating proxy user name

   - Password – authenticating proxy password

   - Proxy Host – URL of HTTP proxy machine.

   - Proxy Port – the port number of the proxy machine.

Then submit the form (see Figure 12). The web application that invokes web services needs this information to make http requests across network fire-walls.

Figure 12: Authentication Details

Known Issues

Web Service Invoker (`wsInvoker.war`)

- The web service invoker does not support sending call back addresses to an asynchronous web service.

- The web service invoker component is not able to invoke web services with complex data types as their parameters.

YAWL XForms Processor (`YAWLXForms.war`)
- Currently forms cannot be generated for schemas that contain enumerated types.

Troubleshooting

I've just finished setting up the YAWL Engine and I don't know my UserID and password. Do I need to request one from the YAWL Administrator?

No. As the person testing or viewing the YAWL Engine, you are welcome to assume the role of Administrator. The Log In details for the Administrator are (case-sensitive):

UserID: `admin`

Password: `YAWL`

Do I have to start the Tomcat service every time I want to use YAWL?

No. Once you start the Tomcat service it will remain started, preserving all your uploaded specifications and the status of "work", until the computer has been shutdown or restarted.

The Tomcat service remains started regardless of whether the Tomcat Monitor has been "exited". If you restart the computer, the Tomcat service will stop upon shutdown, and start once the computer has restarted.

When I navigate to another page in the YAWL Engine, the browser tells me that "The page cannot be displayed".

i) Make sure that the Tomcat service is "started".

ii) Try refreshing the page (F5). YAWL has to wait approximately 10 – 15 seconds for the Tomcat web service to start.

I can't seem to Log In to the YAWL Engine, using the Administrator ID. I keep getting an error message, "Your attempt at login failed try again".

There could be any of three possible answers to this problem. Please check each answer in the following order:

i) Have you met the software requirements of YAWL?

See the Software Requirements section of the Requirements & Installation chapter.

ii) Check the UserID and Password

The Administrator's UserID is **`admin`** and the password is **`YAWL.`** Both are case-sensitive.

iii) Has Tomcat been installed on a port different from the default port value of 8080?

The "`worklist.war`" component is configured to look up the "`engine.war`" component from an address in "`web.xml`".

If you have installed Tomcat on a port other than 8080, then go to the "`web.xml`" file for `worklist.war` and go to the context param named "InterfaceB_BackEnd". Change the value to your port and restart Tomcat.

iv) Check that `yawl.war` is installed. "`worklist.war`" is not a stand alone web application. It needs to "talk" via XML to `yawl.war` to retreive and state and instructions.   See installation instructions to install this component.

Does the YAWL Engine have database support ?

Yes, it is optional, and uses PostgreSQL by default.

Third–Party Software Acknowledgements

 (a) Software developed by the JDOM Project (http://www.jdom.org/),

JDOM Beta 9, Copyright © 2000-2003 Jason Hunter & Brett McLaughlin. All rights reserved. This software is subject to the terms of the licence available at http://www.jdom.org/.

(b) Software developed by the Apache Software Foundation (http://www.apache.org/) (Xerces2 Java Parser 2.6.0 Release) (http://xml.apache.org/xerces2-j/index.html), Copyright © 1999-2003 The Apache Software Foundation. All rights reserved. This software is subject to the terms of the licence available at http://xml.apache.org/.

Web Services Invocation Framework 2.0.- Release (http://ws.apache.org/wsif/), Copyright © 1999-2003 The Apache Software Foundation. All rights reserved. This software is subject to the terms of the licence available at http://xml.apache.org.

(c) The Saxon version 7 XLST Processor from Michael Kay (http://saxon.sourceforge.net/). This software is subject to the Mozilla Public License Version 1.0 and may not be used except in compliance with that License (a copy is available at http://www.mozilla.org/MPL/). The source code version of this software is available under the terms of that License.

(d) Chiba XForms processor version 0.9.9, available at
http://sourceforge.net/projects/chiba/

(e) XSD and EMF (source available at http://www.eclipse.org/xsd)

XSD and EMF are made available under the Common Public License:

```
Common Public License Version 1.0
THE ACCOMPANYING PROGRAM IS PROVIDED UNDER THE TERMS OF THIS COMMON PUBLIC LICENSE
("AGREEMENT"). ANY USE, REPRODUCTION OR DISTRIBUTION OF THE PROGRAM CONSTITUTES
RECIPIENT'S ACCEPTANCE OF THIS AGREEMENT.
1. DEFINITIONS
"Contribution" means:
a) in the case of the initial Contributor, the initial code and documentation
distributed under this Agreement, and
b) in the case of each subsequent Contributor:
i) changes to the Program, and
ii) additions to the Program;
where such changes and/or additions to the Program originate from and are
distributed by that particular Contributor. A Contribution 'originates' from a
Contributor if it was added to the Program by such Contributor itself or anyone
acting on such Contributor's behalf. Contributions do not include additions to the
Program which: (i) are separate modules of software distributed in conjunction
with the Program under their own license agreement, and (ii) are not derivative
works of the Program.
"Contributor" means any person or entity that distributes the Program.
"Licensed Patents " mean patent claims licensable by a Contributor which are
necessarily infringed by the use or sale of its Contribution alone or when
combined with the Program.
"Program" means the Contributions distributed in accordance with this Agreement.
"Recipient" means anyone who receives the Program under this Agreement, including
all Contributors.
2. GRANT OF RIGHTS
a) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free copyright license to reproduce,
prepare derivative works of, publicly display, publicly perform, distribute and
sublicense the Contribution of such Contributor, if any, and such derivative
works, in source code and object code form.
b) Subject to the terms of this Agreement, each Contributor hereby grants
Recipient a non-exclusive, worldwide, royalty-free patent license under Licensed
Patents to make, use, sell, offer to sell, import and otherwise transfer the
Contribution of such Contributor, if any, in source code and object code form.
This patent license shall apply to the combination of the Contribution and the
Program if, at the time the Contribution is added by the Contributor, such
addition of the Contribution causes such combination to be covered by the Licensed
Patents. The patent license shall not apply to any other combinations which
include the Contribution. No hardware per se is licensed hereunder.
c) Recipient understands that although each Contributor grants the licenses to its
Contributions set forth herein, no assurances are provided by any Contributor that
the Program does not infringe the patent or other intellectual property rights of
any other entity. Each Contributor disclaims any liability to Recipient for claims
brought by any other entity based on infringement of intellectual property rights
or otherwise. As a condition to exercising the rights and licenses granted
hereunder, each Recipient hereby assumes sole responsibility to secure any other
intellectual property rights needed, if any. For example, if a third party patent
license is required to allow Recipient to distribute the Program, it is
Recipient's responsibility to acquire that license before distributing the
Program.
d) Each Contributor represents that to its knowledge it has sufficient copyright
rights in its Contribution, if any, to grant the copyright license set forth in
this Agreement.
3. REQUIREMENTS
A Contributor may choose to distribute the Program in object code form under its
own license agreement, provided that:
a) it complies with the terms and conditions of this Agreement; and
b) its license agreement:
i) effectively disclaims on behalf of all Contributors all warranties and
conditions, express and implied, including warranties or conditions of title and
non-infringement, and implied warranties or conditions of merchantability and
fitness for a particular purpose;
```

ii) effectively excludes on behalf of all Contributors all liability for damages, including direct, indirect, special, incidental and consequential damages, such as lost profits;

iii) states that any provisions which differ from this Agreement are offered by that Contributor alone and not by any other party; and

iv) states that source code for the Program is available from such Contributor, and informs licensees how to obtain it in a reasonable manner on or through a medium customarily used for software exchange.

When the Program is made available in source code form:

a) it must be made available under this Agreement; and

b) a copy of this Agreement must be included with each copy of the Program. Contributors may not remove or alter any copyright notices contained within the Program.

Each Contributor must identify itself as the originator of its Contribution, if any, in a manner that reasonably allows subsequent Recipients to identify the originator of the Contribution.

4. COMMERCIAL DISTRIBUTION

Commercial distributors of software may accept certain responsibilities with respect to end users, business partners and the like. While this license is intended to facilitate the commercial use of the Program, the Contributor who includes the Program in a commercial product offering should do so in a manner which does not create potential liability for other Contributors. Therefore, if a Contributor includes the Program in a commercial product offering, such Contributor ("Commercial Contributor") hereby agrees to defend and indemnify every other Contributor ("Indemnified Contributor") against any losses, damages and costs (collectively "Losses") arising from claims, lawsuits and other legal actions brought by a third party against the Indemnified Contributor to the extent caused by the acts or omissions of such Commercial Contributor in connection with its distribution of the Program in a commercial product offering. The obligations in this section do not apply to any claims or Losses relating to any actual or alleged intellectual property infringement. In order to qualify, an Indemnified Contributor must: a) promptly notify the Commercial Contributor in writing of such claim, and b) allow the Commercial Contributor to control, and cooperate with the Commercial Contributor in, the defense and any related settlement negotiations. The Indemnified Contributor may participate in any such claim at its own expense.

For example, a Contributor might include the Program in a commercial product offering, Product X. That Contributor is then a Commercial Contributor. If that Commercial Contributor then makes performance claims, or offers warranties related to Product X, those performance claims and warranties are such Commercial Contributor's responsibility alone. Under this section, the Commercial Contributor would have to defend claims against the other Contributors related to those performance claims and warranties, and if a court requires any other Contributor to pay any damages as a result, the Commercial Contributor must pay those damages.

5. NO WARRANTY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, THE PROGRAM IS PROVIDED ON AN "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, EITHER EXPRESS OR IMPLIED INCLUDING, WITHOUT LIMITATION, ANY WARRANTIES OR CONDITIONS OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Each Recipient is solely responsible for determining the appropriateness of using and distributing the Program and assumes all risks associated with its exercise of rights under this Agreement, including but not limited to the risks and costs of program errors, compliance with applicable laws, damage to or loss of data, programs or equipment, and unavailability or interruption of operations.

6. DISCLAIMER OF LIABILITY

EXCEPT AS EXPRESSLY SET FORTH IN THIS AGREEMENT, NEITHER RECIPIENT NOR ANY CONTRIBUTORS SHALL HAVE ANY LIABILITY FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING WITHOUT LIMITATION LOST PROFITS), HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OR DISTRIBUTION OF THE PROGRAM OR THE EXERCISE OF ANY RIGHTS GRANTED HEREUNDER, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

7. GENERAL

If any provision of this Agreement is invalid or unenforceable under applicable law, it shall not affect the validity or enforceability of the remainder of the terms of this Agreement, and without further action by the parties hereto, such provision shall be reformed to the minimum extent necessary to make such provision valid and enforceable.

If Recipient institutes patent litigation against a Contributor with respect to a patent applicable to software (including a cross-claim or counterclaim in a lawsuit), then any patent licenses granted by that Contributor to such Recipient under this Agreement shall terminate as of the date such litigation is filed. In addition, if Recipient institutes patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Program itself (excluding combinations of the Program with other software or hardware) infringes such Recipient's patent(s), then such Recipient's rights granted under Section 2(b) shall terminate as of the date such litigation is filed.

All Recipient's rights under this Agreement shall terminate if it fails to comply with any of the material terms or conditions of this Agreement and does not cure

such failure in a reasonable period of time after becoming aware of such
noncompliance. If all Recipient's rights under this Agreement terminate, Recipient
agrees to cease use and distribution of the Program as soon as reasonably
practicable. However, Recipient's obligations under this Agreement and any
licenses granted by Recipient relating to the Program shall continue and survive.
Everyone is permitted to copy and distribute copies of this Agreement, but in
order to avoid inconsistency the Agreement is copyrighted and may only be modified
in the following manner. The Agreement Steward reserves the right to publish new
versions (including revisions) of this Agreement from time to time. No one other
than the Agreement Steward has the right to modify this Agreement. IBM is the
initial Agreement Steward. IBM may assign the responsibility to serve as the
Agreement Steward to a suitable separate entity. Each new version of the Agreement
will be given a distinguishing version number. The Program (including
Contributions) may always be distributed subject to the version of the Agreement
under which it was received. In addition, after a new version of the Agreement is
published, Contributor may elect to distribute the Program (including its
Contributions) under the new version. Except as expressly stated in Sections 2(a)
and 2(b) above, Recipient receives no rights or licenses to the intellectual
property of any Contributor under this Agreement, whether expressly, by
implication, estoppel or otherwise. All rights in the Program not expressly
granted under this Agreement are reserved.
This Agreement is governed by the laws of the State of New York and the
intellectual property laws of the United States of America. No party to this
Agreement will bring a legal action under this Agreement more than one year after
the cause of action arose. Each party waives its rights to a jury trial in any
resulting litigation.

Acknowledgements

This documentation includes feedback from Lachlan Aldred and Marlon Dumas,
Queensland University of Technology.